Esraa AL-Hafidh "2013"

أنواع الـ Sorting :

1) Selektion Sort :-

* المزايا :-

1) simple implemention ← سهل التنفيذ
2) Simple algorithm ← طريقة سهل غير معقدة مثل باقي الأنواع الأخرى

يستخدم في حالة معينة ان يكون هناك Array صغير نسبياً أي متكونة من 10 او 50 او حتى 100 عنصر

3) لا تحتاج الى ذاكرة (Memory) زيادة

* العيوب

1) بطيئ لانه هو خوارزمية اي Algorthem of n^2 ← $O(n^2)$
2) Blind Algorthem اي ما معناه الخوارزمية العمياء السبب وحدوهما ان في حالة استخدام هذه الطريقة او النوع تجب عليه المرور بجميع العناصر حتى وان حصلنا على النتيجة المطلوبة، سيقوم هذا النوع من Sorting بالمرور على جميع العناصر بالكامل، ولهذا قلنا ان هذا النوع مناسب لـ Arrays التي تكون صغيرة الحجم.

مثال :

مطلوب ترتيب العناصر المعطاة من الاصغر الى الاكبر

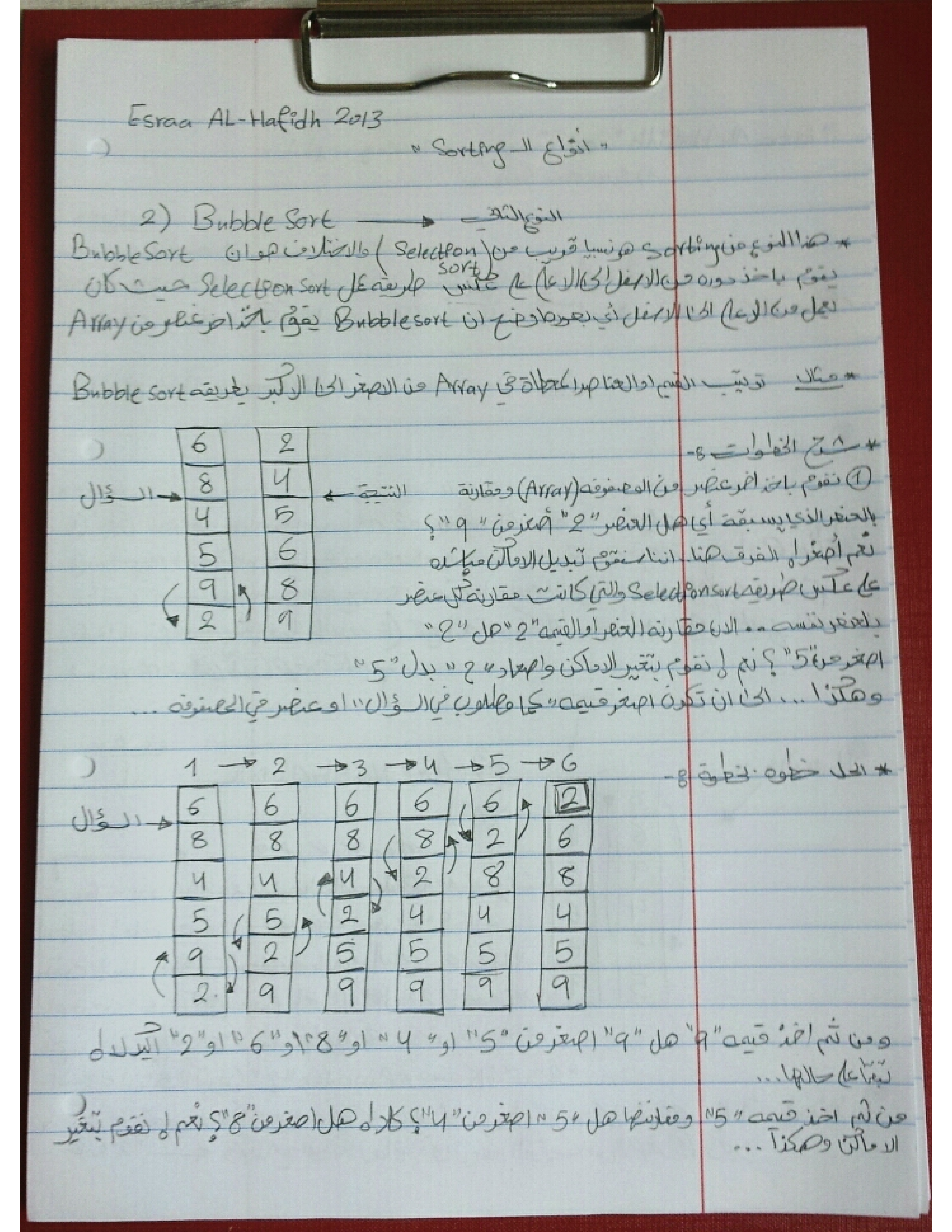
$O(n^2)$

min = ~~8~~ ~~6~~ ~~4~~ 2

السؤال المعطاة	النتيجة
8	2
6	4
9	5
4	6
2	8
5	9

نقوم بمقارنة العنصر الاول "8" ببقية العناصر وعليه هل 8 اصغر من 6؟ الاجابة لا، فاخذ قيمة الـ"6" هل "9" اصغر "6"؟ كلا، نقوم بابقاء العنصر "6" ومقارنته بالعنصر التالي ... هل "4" اصغر من "6"؟ نعم نقوم باخذ "4" هل "2" اصغر من "4"؟ نعم ... نقوم باخذ "2" ومقارنتها مع العنصر الاخير وهو "5" والنتيجة ان "5" اكبر من "2" وعليه ان اصغر قيمة كانت "2"

* نقوم بتغيير الاماكن بين عنصر "8" وعنصر "2" أي ان "2" ستكون اول عنصر في القائمة والعنصر "8" بدل مكان "2" ← وهكذا مع بقية العناصر الى ان تم الترتيب من الاصغر الى الاكبر



Esraa AL-Hafidh 2013

« انواع الـ Sorting »

2) Bubble Sort → النوع الثاني

* هذا النوع من Sorting هو نسبيا قريب من (Selection sort) فالاختلاف هو ان Bubble Sort يقوم باخذ دوره من الاسفل الى الاعلى على عكس طريقة عمل Selection sort حيث كان تعمل من الاعلى الى الاسفل أي بمعنى اوضح ان Bubble sort يقوم باخذ اخر عنصر من Array

* مثال: ترتيب القيم او العناصر المعطاة في Array من الاصغر الى الاكبر بطريقة Bubble sort

السؤال	النتيجة
6	2
8	4
4	5
5	6
9	8
2	9

* شرح الخطوات :-

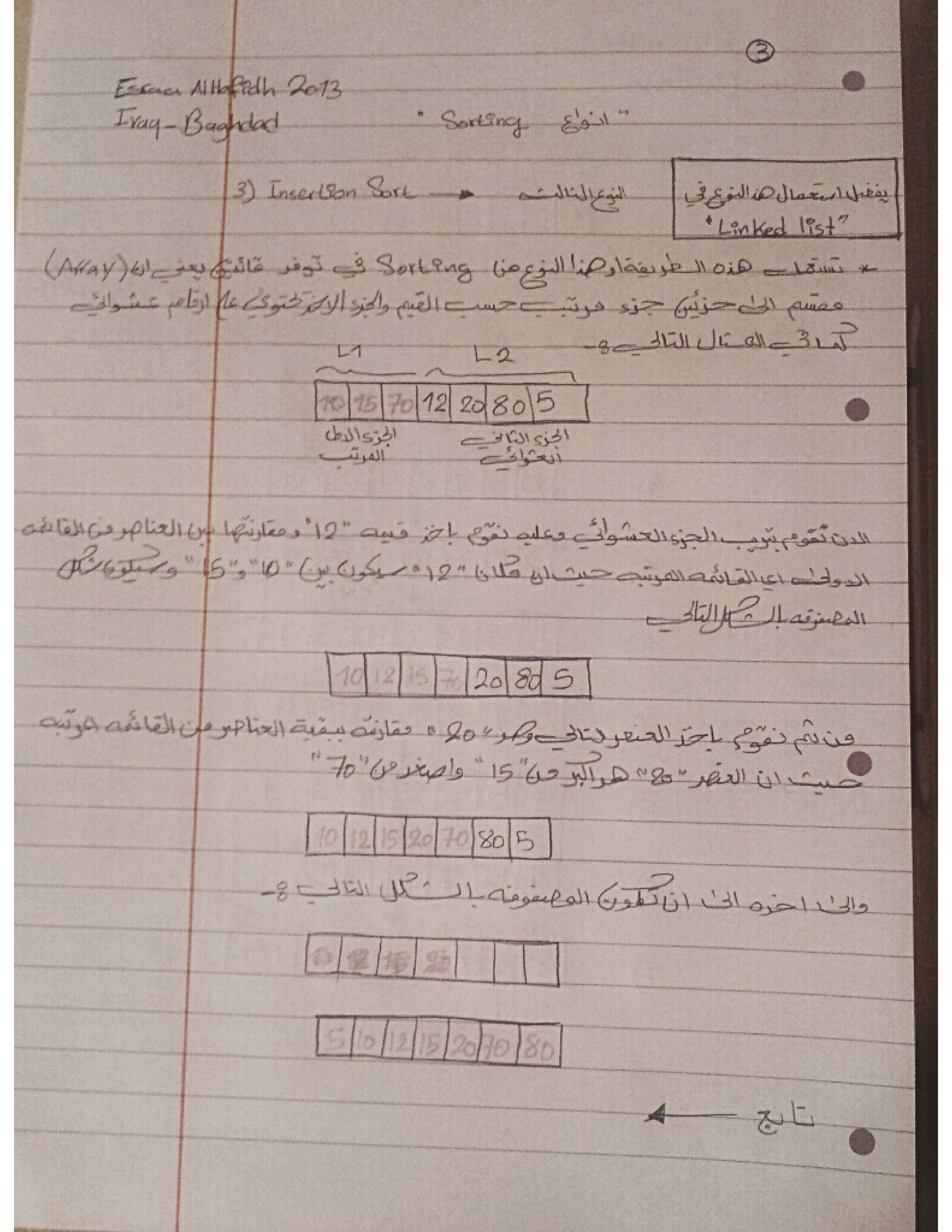
① نقوم باخذ اخر عنصر من المصفوفة (Array) ومقارنة بالعنصر الذي يسبقه أي هل العنصر "2" أصغر من "9"؟ نعم أصغر، الفرق هنا اننا نقوم تبديل الاماكن مباشرة على عكس طريقة Selection sort والتي كانت مقارنة كل عنصر بالعنصر نفسه .. الان مقارنة العنصر او القيمة "2" هل "2" اصغر من "5"؟ نعم إذا نقوم بتغيير الاماكن واصعاد "2" بدل "5" وهكذا ... الى ان تكون اصغر قيمة "كما مطلوب في السؤال" اول عنصر في المصفوفة ...

* الحل خطوة بخطوة :-

1 (السؤال)	2	3	4	5	6
6	6	6	6	6	2
8	8	8	8	2	6
4	4	4	2	8	8
5	5	2	4	4	4
9	2	5	5	5	5
2	9	9	9	9	9

ومن ثم اخذ قيمة "9" هل "9" اصغر من "5" او "4" او "8" او "6" او "2" الجواب لا تبقى على حالها ...

من ثم اخذ قيمة "5" ومقارنتها هل "5" اصغر من "4"؟ كلا، هل اصغر من "8"؟ نعم إذا نقوم بتغيير الاماكن وهكذا ...



Esraa AlHafidh 2013
Iraq-Baghdad

" انواع Sorting "

3) Insertion Sort ← النوع الثالث

يفضل استعمال هذا النوع في "Linked list"

* تستعمل هذه الطريقة او هذا النوع من Sorting في توفر قائمة يعني ان (Array) مقسم الى جزئين جزء مرتب حسب القيم والجزء الاخر يحتوي على ارقام عشوائية كما في المثال التالي :-

L1 | L2

10	15	70	12	20	80	5

الجزء الاول المرتب — الجزء الثاني العشوائي

الان نقوم بترتيب الجزء العشوائي وعليه نقوم باخذ قيمة "12" ومقارنتها بين العناصر من القائمة الاولى اي القائمة المرتبة حيث ان مثلا "12" سيكون بين "10" و "15" وسيكون شكل المصفوفة بالشكل التالي

10	12	15	70	20	80	5

ومن ثم نقوم باخذ العنصر التالي وهو "20" ومقارنته ببقية العناصر من القائمة مرتبة حيث ان العنصر "20" هو اكبر من "15" واصغر من "70"

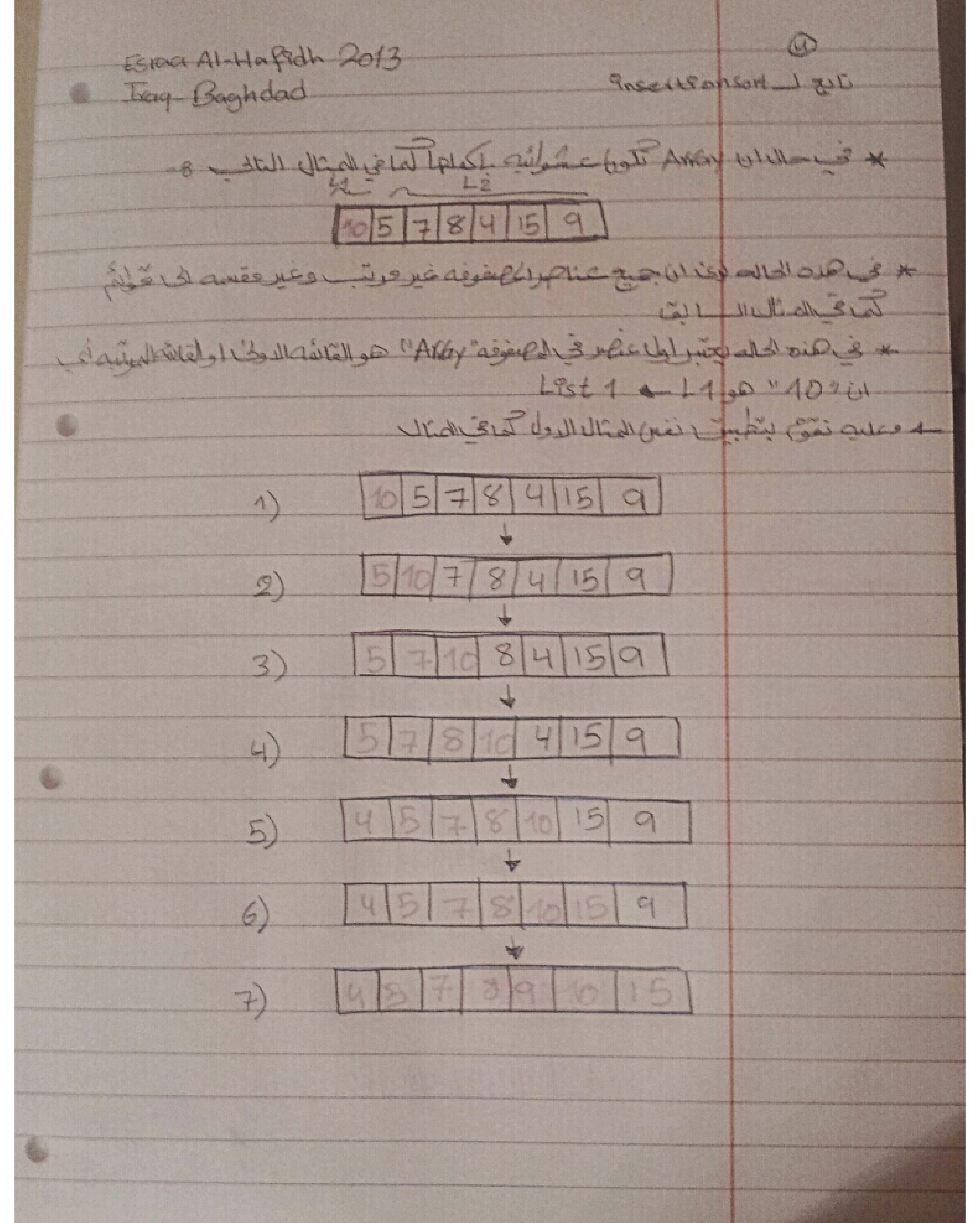
10	12	15	20	70	80	5

والى اخره الى ان تكون المصفوفة بالشكل التالي :-

[illegible]	[illegible]	[illegible]	[illegible]			

5	10	12	15	20	70	80

ناتج ←



Esraa Al-Hafidh 2013
Iraq-Baghdad

تابع لـ Insertionsort

* في حال ان Array تكون عشوائية بكاملها كما في المثال التالي :-

L1   L2

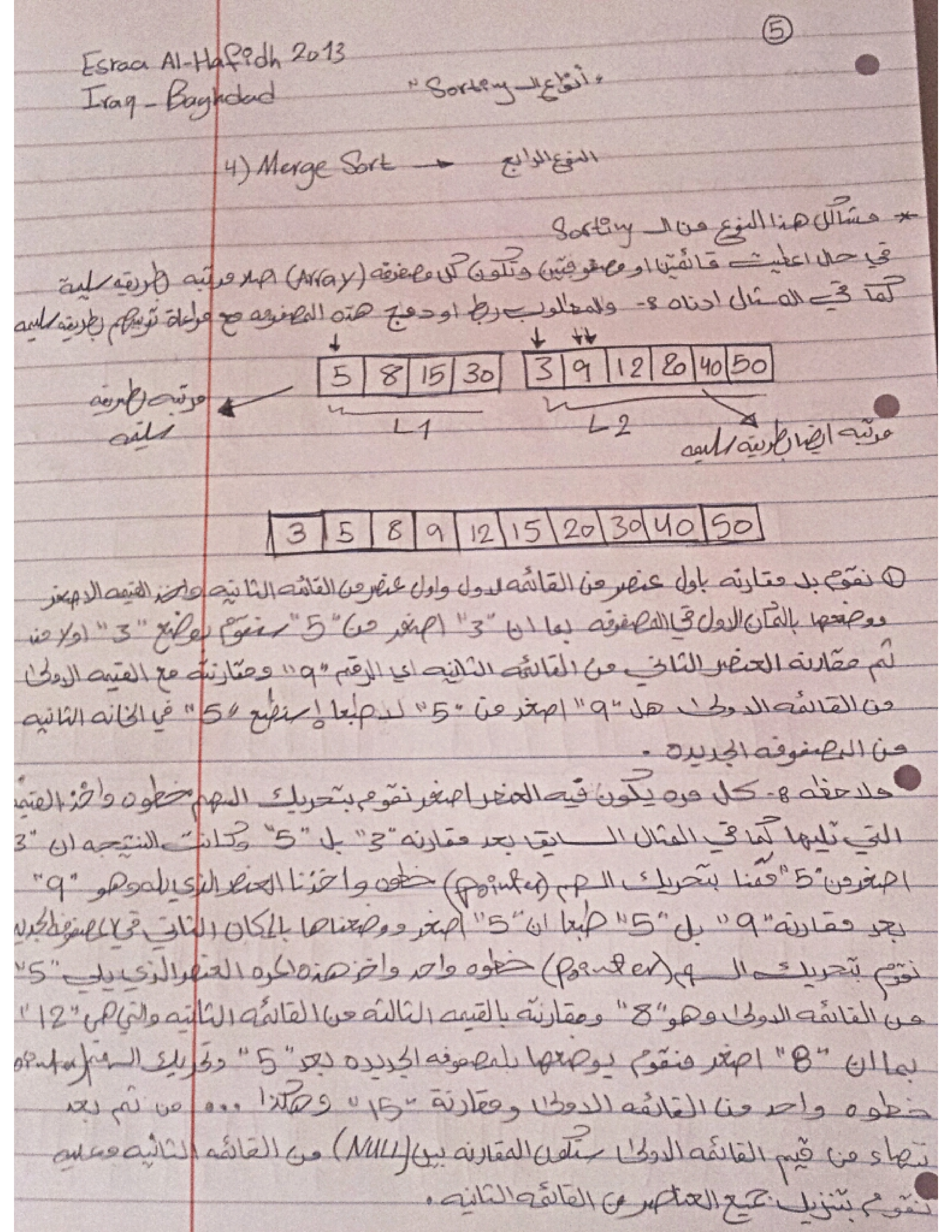
10	5	7	8	4	15	9

* في هذه الحالة نجد ان جميع عناصر المصفوفة غير مرتبة وغير مقسمة الى قوائم كما في المثال السابق

* في هذه الحالة يعتبر اول عنصر في المصفوفة "Array" هو القائمة الاولى او القائمة المرتبة أي ان "10" هو L1 ← List 1

← وعليه نقوم بتطبيق نفس المثال الاول كما في المثال

1)	10	5	7	8	4	15	9
2)	5	10	7	8	4	15	9
3)	5	7	10	8	4	15	9
4)	5	7	8	10	4	15	9
5)	4	5	7	8	10	15	9
6)	4	5	7	8	10	15	9
7)	4	5	7	8	9	10	15



Esraa Al-Hafidh 2013
Iraq - Baghdad

أنواع الـ "Sorting"

4) Merge Sort ← النوع الرابع

* مشاكل هذا النوع من الـ Sorting
في حال اعطيت قائمتين او مصفوفتين وتكون كل مصفوفة (Array) هي مرتبة بطريقة سليمة كما في المثال ادناه :- والمطلوب ربط او دمج هذه المصفوفة مع مراعاة ترتيبهم بطريقة سليمة

5	8	15	30

L1 — مرتبة بطريقة سليمة

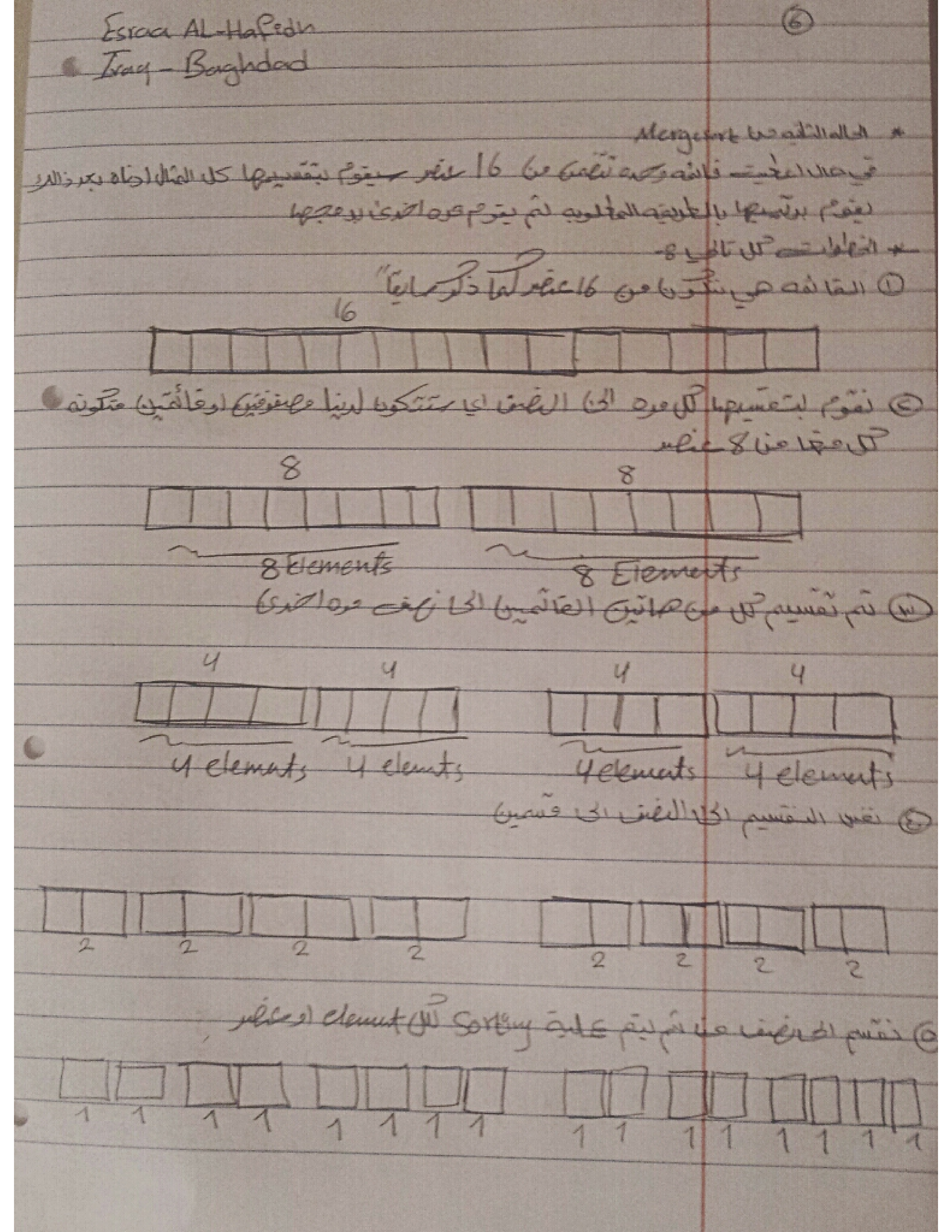
3	9	12	20	40	50

L2 — مرتبة ايضا بطريقة سليمة

3	5	8	9	12	15	20	30	40	50

① نقوم بل مقارنة باول عنصر من القائمة الاولى واول عنصر من القائمة الثانية واخذ القيمة الاصغر ووضعها بالمكان الاول في المصفوفة بما ان "3" اصغر من "5" سنقوم بوضع "3" اولا ومن ثم مقارنة العنصر الثاني من القائمة الثانية اي الرقم "9" ومقارنته مع القيمة الاولى من القائمة الاولى هل "9" اصغر من "5" لا طبعا إ سنضع "5" في الخانة الثانية من المصفوفة الجديدة .

ملاحظة :- كل مرة يكون فيه العنصر اصغر نقوم بتحريك السهم خطوة واخذ القيمة التي تليها كما في المثال السابق بعد مقارنة "3" بل "5" وكانت النتيجة ان "3" اصغر من "5" قمنا بتحريك السهم (Pointer) خطوة واخذنا العنصر الذي يليه وهو "9" بعد مقارنة "9" بل "5" طبعا ان "5" اصغر ووضعناها بالمكان الثاني في المصفوفة الجديدة نقوم بتحريك السهم (Pointer) خطوة واحد واخذ هذه المرة العنصر الذي يلي "5" من القائمة الاولى وهو "8" ومقارنته بالقيمة الثالثة من القائمة الثانية والتي هي "12" بما ان "8" اصغر فنقوم بوضعها بالمصفوفة الجديدة بعد "5" وتحريك السهم (Pointer) خطوة واحد من القائمة الاولى ومقارنة "15" وهكذا ... ومن ثم بعد نفاد من قيم القائمة الاولى ستكون المقارنة بين (NULL) من القائمة الثانية وعليه نقوم بتنزيل جميع العناصر من القائمة الثانية .



Esraa AL-Hafidh
Iraq - Baghdad

* الحالة الثانية من Mergesort

في حال الخطيئة قائمة وحيدة تتضمن من 16 عنصر سيقوم بتقسيمها كل المثال اعلاه بعد ذلك يقوم بترتيبها بالطريقة المطلوبة ثم يترجم مرة اخرى بدمجها

* الخطوات كل تالي :-

① القائمة هي تتكون من 16 عنصر كما ذكر سابقاً"

16

② نقوم بتقسيمها كل مرة الى النصف اي ستتكون لدينا مصفوفتين او قائمتين متكونة كل منها من 8 عناصر

8 8

8 Elements 8 Elements

③ ثم تقسيم كل من هاتين القائمتين الى نصف مرة اخرى

4 4 4 4

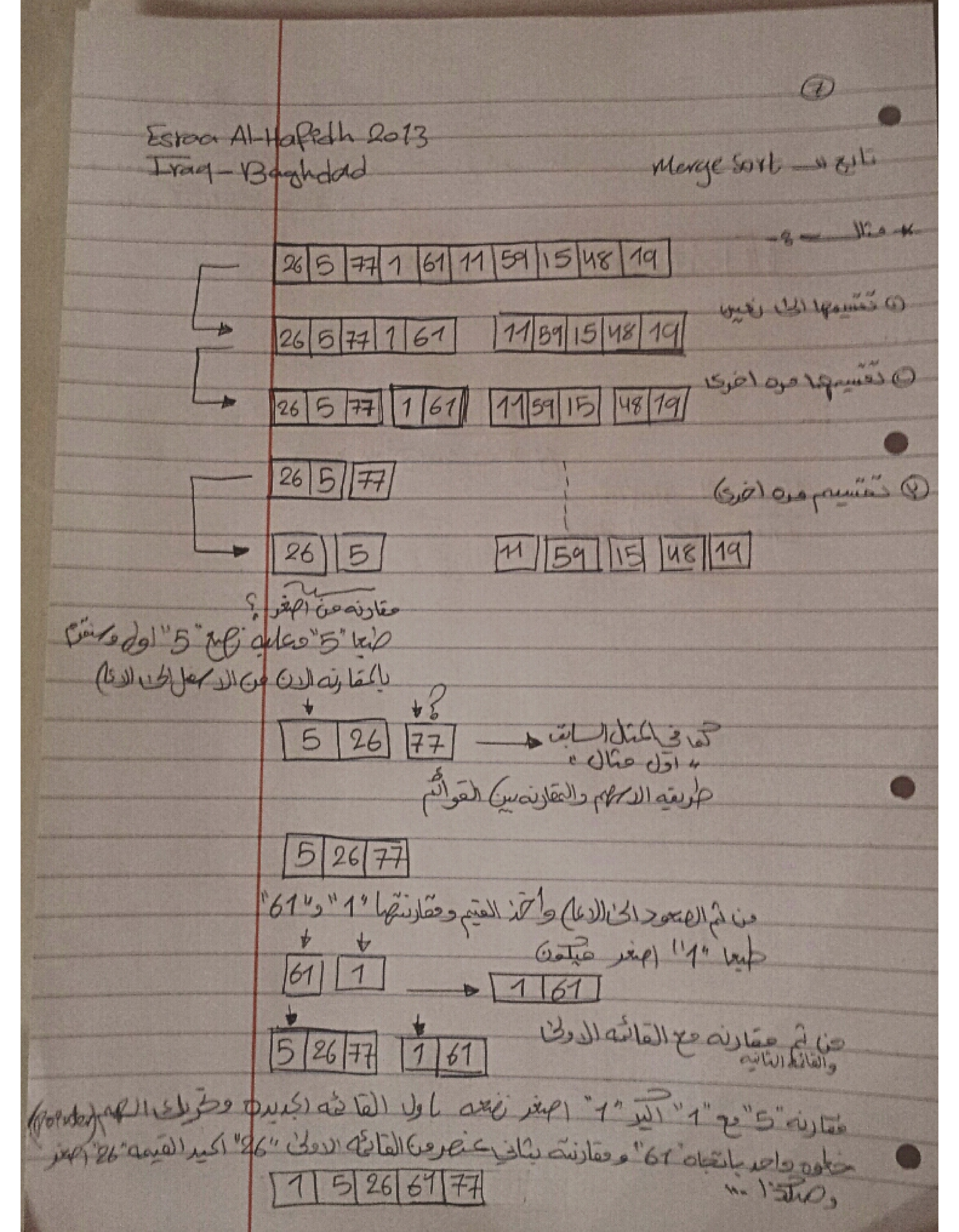
4 elements 4 elements 4 elements 4 elements

④ نفس التقسيم الى النصف الى قسمين

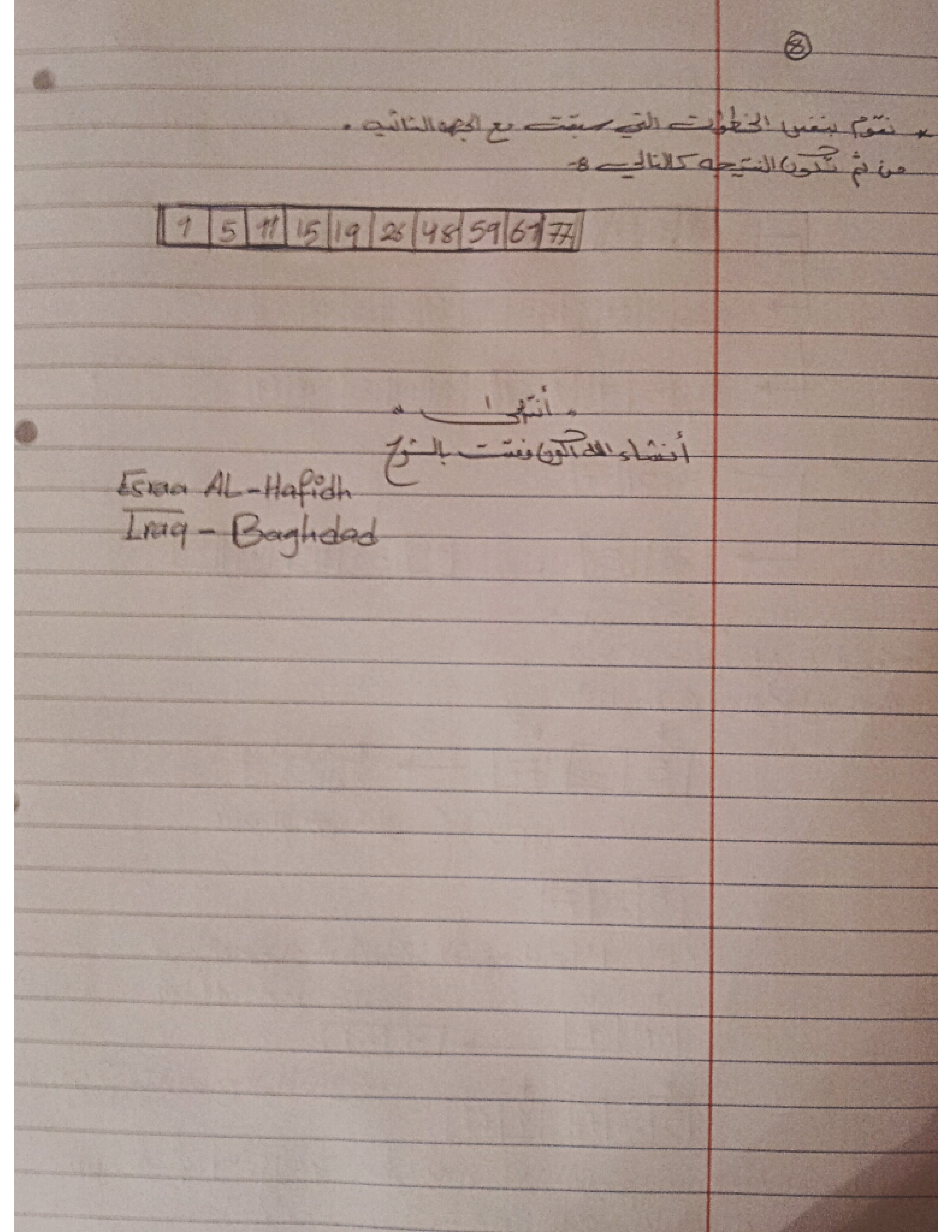
2 2 2 2 2 2 2 2

⑤ نقسم الى النصف حتى ثم يتم عملية sorting لكل element او عنصر

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1



(7)
Esraa Al-Hafidh 2013
Iraq-Baghdad
تابع — Merge sort
مثال :-
26 | 5 | 77 | 1 | 61 | 11 | 59 | 15 | 48 | 19
① تقسيمها الى نصفين
26 | 5 | 77 | 1 | 61
11 | 59 | 15 | 48 | 19
② تقسيمها مرة اخرى
26 | 5 | 77
1 | 61
11 | 59 | 15
48 | 19
③ تقسيم مرة اخرى
26 | 5 | 77
26
5
11
59
15
48
19
مقارنة من الاصغر؟
طبعا "5" فعليه نضع "5" اول ونستمر بالمقارنة الان من الاسفل الى الاعلى
5 | 26
77
كما في المثال السابق "اول مثال" طريقة الاسهم والمقارنة بين القوائم
5 | 26 | 77
من ثم الصعود الى الاعلى واخذ القيم ومقارنتها "1" و "61"
طبعا "1" اصغر فيكون
61
1
1 | 61
5 | 26 | 77
1 | 61
من ثم مقارنة مع القائمة الاولى والقائمة الثانية
مقارنة "5" مع "1" الاكبر "1" اصغر نضعه باول القائمة الجديدة وتحريك السهم (pointer) خطوة واحدة باتجاه "61" ومقارنته بثاني عنصر من القائمة الاولى "26" الاكبر القيمة "26" اصغر
1 | 5 | 26 | 61 | 77
وهكذا...



* نقوم بنفس الخطوات التي سبقت مع الجهة الثانية .

من ثم تكون النتيجة كالتالي :-

1	5	11	15	19	26	48	59	61	77

انتهى .

أنشاء الله تكونون فهمتم بالشرح

Esraa AL-Hafidh

Iraq - Baghdad